electronics

MDPI

Review

Comparative Analysis of Passkeys (FIDO2 Authentication) on Android and iOS for GDPR Compliance in Biometric Data Protection

Albert Carroll and Shahram Latifi *

Department of Electrical & Computer Engineering, University of Nevada, Las Vegas, NV 89154, USA; carro120@unlv.nevada.edu

* Correspondence: shahram.latifi@unlv.edu

Abstract

Biometric authentication, such as facial recognition and fingerprint scanning, is now standard on mobile devices, offering secure and convenient access. However, the processing of biometric data is tightly regulated under the European Union's General Data Protection Regulation (GDPR), where such data qualifies as "special category" personal data when used for uniquely identifying individuals. Compliance requires meeting strict conditions, including explicit consent and data protection by design. Passkeys, the modern name for FIDO2-based authentication credentials developed by the FIDO Alliance, enable passwordless login using public key cryptography. Its "match-on-device" architecture stores biometric data locally in secure hardware (e.g., Android's Trusted Execution Environment, Apple's Secure Enclave), potentially reducing the regulatory obligations associated with cloud-based biometric processing. This paper examines how Passkeys are implemented on Android and iOS platforms and their differences in architecture, API access, and hardware design, and how those differences affect compliance with the GDPR. Through a comparative analysis, we evaluate the extent to which each platform supports local processing, data minimization, and user control—key principles under GDPR. We find that while both platforms implement strong local protections, differences in developer access, trust models, and biometric isolation can influence the effectiveness and regulatory exposure of Passkeys deployment. These differences have direct implications for privacy risk, legal compliance, and implementation choices by app developers and service providers. Our findings highlight the need for platform-aware design and regulatory interpretation in the deployment of biometric authentication technologies. This work can help inform stakeholders, policymakers, and legal experts in drafting robust privacy and ethical policies—not only in the realm of biometrics but across AI technologies more broadly. By understanding platform-level implications, future frameworks can better align technical design with regulatory compliance and ethical standards.

Keywords: android; biometric data protection; FIDO2; GDPR; mobile security; Passkeys (device-bound); iOS

check for updates

Academic Editor: Gongping Yang

Received: 25 August 2025
Revised: 30 September 2025
Accepted: 2 October 2025
Published: 13 October 2025

Citation: Carroll, A.; Latifi, S. Comparative Analysis of Passkeys (FIDO2 Authentication) on Android and iOS for GDPR Compliance in Biometric Data Protection. *Electronics* **2025**, *14*, 4018. https://doi.org/10.3390/electronics14204018

Copyright: © 2025 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (https://creativecommons.org/licenses/by/4.0/).

1. Introduction

Biometric authentication is now widely used on mobile devices, providing fast verification through fingerprint, face recognition, voice recognition, typing and touch gestures [1,2]. These technologies allow individuals to bypass traditional methods like passwords or PINs

with a simple glance or touch. However, the handling of biometric data on mobile devices introduces significant privacy and security concerns. This includes risks of data breaches, spoofing attacks, and regulatory issues [3]. The European Union's GDPR addresses these concerns by imposing strict rules on biometric data processing. Under Article 9(1), biometric data is classified as 'special category' when used for unique identification. In such cases, explicit consent or other exemptions are required [4]. Even when special category status does not apply, the GDPR imposes general obligations for data transfer, including data minimization (Article 5), lawfulness of processing (Article 6), "privacy-by-design and by-default" (Article 25), and strong security measures for processing (Article 32) [4].

This paper argues that although both Android and iOS implement Passkeys consistently with GDPR principles, iOS offers more uniform compliance through its integrated hardware–software model, while Android's fragmented ecosystem introduces variability in security and greater regulatory risk. Device-bound Passkeys, built on the FIDO2 standard, directly address many of these concerns. They enable secure, passwordless login on mobile devices through FIDO2 protocols (WebAuthn and CTAP2) [5–9]. Passkey functionality is now built into most platforms, and its foundation in the FIDO2 standard relies on a match-on-device model in which both the cryptographic keys and the biometric templates that unlock them remain entirely within secure hardware modules on the user's device—such as Android's Trusted Execution Environment (TEE) or Apple's Secure Enclave [10–12]. Passkeys rely on cryptography to authenticate users without transmitting biometric data to external servers. Biometric matching occurs exclusively on-device, preventing the transmission of biometric data to servers or third parties. When performed in this way, Passkeys may avoid triggering GDPR special or "sensitive" category obligations, thereby reducing regulatory burdens for app developers and service providers [10,13].

Android and iOS, which dominate the global mobile market, differ significantly in how they manage biometric data. On Android, biometric authentication typically uses the BiometricPrompt API, which standardizes user interactions and biometric processing [11]. The TEE on Android devices provides hardware-backed isolation for sensitive operations, while newer devices incorporate StrongBox, a hardware security module that enhances cryptographic key storage [14–16]. However, Android's open ecosystem and manufacturer diversity create fragmentation, security risks, and regulatory compliance, leading to inconsistencies in Passkeys quality and biometric security [14,17]. On the other hand, Apple's iOS is a uniform hardware–software model with all devices similar in design and hardware. Biometric authentication is governed by the LocalAuthentication API framework, with biometric templates isolated within the Secure Enclave, a tamper-resistant security coprocessor [12,18,19]. iOS devices support Passkeys primarily through WebAuthn integration in Safari, with the Secure Enclave handling private key storage and biometric verification. Additionally, iOS also allows the use of external FIDO2-certified security keys via NFC (Near Field Communication), Lightning, or USB connections [20–24]. While Apple's controlled environment may offer stronger baseline privacy protections, it also limits third-party access to the Secure Enclave and prevents apps from directly interacting with biometric templates, which can restrict Passkeys' full adoption across third-party applications.

Given these design and security differences, it is essential to evaluate how Android and iOS implement Passkeys in relation to GDPR requirements for biometric data protection. While Passkeys' architecture emphasizes privacy through decentralization and local processing, the extent to which each platform fulfills this can vary. By examining platform-level differences, this paper identifies the extent to which Android and iOS uphold GDPR principles in their handling of biometric data through Passkeys.

For clarity, this paper distinguishes between device-bound Passkeys (also referred to as simply "Passkeys"), which remain local to a single device in accordance with the FIDO2 standard, and synced passkeys, Apple's implementation that synchronizes credentials across devices via iCloud Keychain. The remainder of this paper is organized as follows: Section 2 outlines the methodology. Section 3 reviews the results, including GDPR requirements for biometric data processing, a technical overview of Passkeys on mobile devices, and an analysis of Android and iOS implementations. Section 4 offers a comparative evaluation, and Section 5 concludes with key findings.

2. Methodology

To evaluate how Android and iOS implement Passkeys in the context of the GDPR, this paper uses a comparative legal and technical approach. The analysis combines a review of key GDPR requirements with an examination of how Passkey authentication is performed on each mobile platform. This method ensures that both regulatory obligations and technical specifications are considered when analyzing how effectively Android and iOS protect biometric data in terms of GDPR principles. The study follows a narrative legal-technical review. Sources included official Android/iOS documentation, FIDO Alliance standards, GDPR text, and peer-reviewed articles. Inclusion required authoritative legal or technical relevance to Passkeys and GDPR; outdated or non-authoritative sources were excluded. The paper focuses on Passkeys integration on Android and iOS, representing the dominant mobile operating systems worldwide. Android holds approximately 70% of the global market share, with iOS capturing roughly 28% [25]. The methodology relies on publicly available information and technical documentation. No direct device testing is performed.

2.1. Legal Evaluation

The GDPR, Regulation (EU) 2016/679, provides the primary legal foundation for evaluating how biometric data is processed within the European Union (EU) and European Economic Area (EEA). This paper focuses on key GDPR articles for personal data, specifically biometric data. Article 5 establishes principles of data processing, including data minimization and storage limitation [4]. Article 6 sets out the lawful basis for processing personal data, such as explicit user consent for biometric use [4]. Article 25 requires data protection by design and by default, requiring systems to incorporate privacy features during development [4]. Finally, Article 32 addresses security of processing, requiring technical and organizational measures, including encryption and proper safeguards [4].

2.2. Technical Evaluation

To look into platform-specific support for GDPR requirements, we define three technical evaluation areas based on GDPR principles and Passkeys specifications. The first area, system architecture, evaluates whether each platform follows a decentralized authentication model by ensuring biometric templates and cryptographic keys remain stored locally on the user's device. This considers protections against unauthorized transmission or off-device storage, in line with GDPR principles such as data minimization and privacy-by-design (Articles 5 and 25) [4]. The second area concerns application programming interfaces (APIs). This examines each platform's native support for Passkeys integration, including the use of secure APIs and user control that govern biometric authentication such as Android's BiometricPrompt, or iOS LocalAuthentication. The evaluation also considers systems for obtaining valid user consent, biometric enrollment processes, and users control over biometric data processing, which reflect GDPR requirements under Articles 6, 25, and 32 [4]. The third and final area is hardware security. This area assesses whether platforms

provide secure key generation, storage, and isolation through hardware-based protections such as the Trusted Execution Environment (TEE) and StrongBox on Android or the Secure Enclave on iOS. Proper key management and hardware-backed isolation are essential for complying with GDPR security obligations under Articles 25 and 32 [4].

2.3. Comparison

These articles and technical analysis provide the legal basis for evaluating how Passkeys work on Android and iOS, handle biometric security and meet GDPR compliance requirements. This also emphasizes minimizing unnecessary data transfers, ensuring local storage, and maintaining secure system design, all of which comply with FIDO's privacy-aware design [26]. This study adopts a narrative legal-technical review rather than an empirical assessment, relying on official documentation, legal texts, and peer-reviewed studies. The comparison highlights how each platform addresses the GDPR's main obligations, whether biometric templates and cryptographic keys remain locally stored and processed (Article 5); whether users are given meaningful consent and control over biometric enrollment, authentication, and deletion (Article 6); whether the overall system architecture embodies privacy-by-design and by default, limiting data exposure and restricting third-party access (Article 25); and whether technical safeguards, including hardware-backed protections such as the Secure Enclave, TEE, or StrongBox, are employed to ensure security of processing (Article 32) [4]. The comparison specifically focuses on platform-specific conditions. On Android, evaluation centers on devices supporting BiometricPrompt and hardware-backed StrongBox, models like Samsung Galaxy series. Android's open-source model introduces variability in security implementations across devices. On iOS, the focus is on devices from iPhone X and newer models equipped with Face ID or Touch ID, utilizing the Secure Enclave for biometric isolation. iOS maintains stricter control over third-party hardware and software access, with Passkeys primarily supported via WebAuthn in Safari or through certified external authenticators.

3. Legal Compliance: GDPR Requirements for Biometric Data Processing

3.1. GDPR Classification of Biometric Data

Biometric data represents one of the most sensitive categories of personal information due to its permanence, uniqueness, and direct links to an individual's physical identity [27]. Unlike passwords or credentials, biometric traits are permanent and irrevocable and cannot be changed once compromised [27]. Recognizing these risks, the GDPR imposes strict legal requirements for the processing of biometric data within the EU and EEA. Article 4(14) of the GDPR classifies biometric data as a form of personal data that arises from technical processing of an individual's physiological or behavioral traits when confirming or establishing the person's identity. Under Article 9(1), biometric data qualifies as a special category of personal data only when it is processed for the purpose of uniquely identifying a person. This distinction is critical, as the processing of special category data is generally prohibited under the GDPR unless a valid exemption applies, most commonly, the explicit consent of data processing under Article 9(2)(a). In this context, "prohibited" means that collecting, storing, analyzing, or otherwise processing special category data is unlawful by default, unless one of the specific legal conditions outlined in Article 9(2) is satisfied. If no valid exemption applies, processing biometric data is strictly forbidden.

However, biometric data used solely for technical functions or local device-level authentication may avoid special category classification, provided the data remains isolated and is not connected to broader identity systems [28]. In such cases, the biometric data is treated as ordinary personal data under the GDPR, subject to general obligations but not the heightened protections of Article 9. GDPR interpretation emphasizes that not all

biometric processing automatically qualifies as special category data. A situation where special category protections may not apply is local-only, isolated biometric processing. Biometric processing may avoid classification as special category data under the GDPR, when it remains strictly local to the device, serves purely technical purposes (such as unlocking the device or securing cryptographic keys), and is not linked to broader identity systems [10,29].

3.2. Core GDPR Principles for Data Processing

Regardless of whether special category status applies, all data processing including biometric data must comply with the general obligations outlined in the GDPR. Article 5 establishes the principle of data minimization, requiring that personal data collection be "adequate, relevant, and limited to what is necessary" for the intended purpose. Applied to biometrics, this principle prohibits over-collection and mandates that only the minimum amount of biometric information necessary for authentication be processed [4].

Under Article 6, all personal data processing requires a lawful basis, such as user consent. Consent must meet strict requirements: freely given, specific, and unambiguous (Article 7(1)), clear information and plain language (Article 7(2)), easy withdrawal without negative consequences (Article 7(3)), and no conditional consent is allowed for unnecessary data (Article 7(4)) [4]. Failure to obtain valid consent makes biometric processing unlawful. Organizations can face fines of up to €20 million or 4% of global annual turnover, along with bans on processing data, data deletion orders, or lawsuits under GDPR based on Articles 83 and 84 [4].

Article 25 reinforces the importance of privacy-by-design and by default. Although the GDPR does not explicitly mandate local processing, Articles 5(1)(f), 25, and 32 emphasize integrity, confidentiality, and privacy-by-design, encouraging system designs to minimize unnecessary data collection, storage, and transmission [4]. Local processing of biometric data, where templates remain securely on the user's device, significantly reduces risks of interception, unauthorized access, and data breaches. Article 25 makes it clear that privacy protections must be built into system design from the start. This includes using isolated biometric processing environments, limiting data sharing outside the device [4].

Finally, Article 32 addresses the security of processing. Both data controllers and data processors are required to implement technical and organizational measures that ensure a level of security appropriate to the risks posed by the processing of personal data. These measures include pseudonymization and encryption of personal data (Article 32(1)(a)), confidentiality, integrity, availability, and resilience of processing systems (Article 32(1)(b)), timely restoration of access to personal data after incidents (Article 32(1)(c)), regular testing and evaluation of security measures (Article 32(1)(d)) [4].

The GDPR sets clear rules on how biometric data can be collected, used, and stored. Biometric processing that enables unique identification, especially when linked to an external identity system, triggers stricter protections under Article 9. In contrast, on-device biometric authentication that keeps data isolated may avoid special category classification and reduce regulatory burdens. That said, all biometric data still falls under GDPR's general requirements, including lawful processing, data minimization, privacy-by-design, and technical security. Passkeys, when properly integrated on these platforms, support compliance with these GDPR principles by enabling local-only biometric processing and secure data management.

4. Technical Overview of Passkeys on Mobile Devices

Passkeys are built on the FIDO2 authentication standard that was developed by the FIDO (Fast IDentity Online) Alliance in partnership with the World Wide Web Consortium

(W3C) to enable passwordless, phishing-resistant user authentication across devices and the web [6,9]. In this section, "Passkeys" refers to the standard, device-bound FIDO2 model. Apple's synced passkeys are analyzed separately in Section 6. Unlike traditional authentication systems that rely on usernames, passwords, and PINs, Passkeys allow a user to sign in to apps and websites with the same process that they use to unlock their device, such as biometric authentication [5]. Passkeys support is already integrated into most modern platforms, including Android (version 9.0 and above), iOS (version 13.0 and above), and major browsers such as Chrome, Safari, Edge, and Firefox. While Passkeys are integrated across most modern devices, the level of privacy protection depends on both software version and hardware capabilities, particularly for biometric data handling.

4.1. FIDO2 Protocols

Passkeys operate on two core protocols defined by the FIDO2 standard. The first protocol, WebAuthn (Web Authentication API), is an extension that allows browsers and applications to interact with authenticators in a secure manner. This registration and authentication extension enables the platform to use a user verification method, allowing the authenticator to return information to the WebAuthn Relying Party with secure verification factors during operation [30]. WebAuthn creates a unique public–private key pair for each relying party, ensuring domain-specific credentials and strong phishing resistance. Authentication uses a local biometric or PIN to unlock the private key, which signs a server challenge while never leaving the device. This architecture directly supports GDPR principles of data minimization and confidentiality [30].

The second protocol is CTAP2 (Client to Authenticator Protocol version 2), a FIDO2 protocol that defines how external or platform authenticators, such as security keys or smartphones, communicate with clients like browsers or apps [7]. This enables the user to authenticate on one device by interacting with an authenticator on another. CTAP2 supports multiple connection methods, including Bluetooth, NFC, and USB, enabling both platform and roaming authenticators [7]. It ensures encrypted communication between the client and authenticator, protecting against relay attacks and unauthorized interception [7].

Passkeys provide secure, user-friendly, and privacy-preserving authentication by combining two key protocols from FIDO2 standards, WebAuthn and CTAP2 [31]. These protocols enable Passkeys to operate through three primary components: the Relying Party (the web service requesting authentication), the Platform or Browser (which communicates with authenticators via the WebAuthn API), and the Authenticator (a secure device that stores private keys and may include biometric sensors). Passkeys' core strength lies in combining robust cryptography with strong privacy protections tailored for mobile devices. Each service (RP) generates a unique asymmetric key pair, preventing password reuse. Credential keys are created for one website and cannot be used elsewhere, providing strong phishing resistance. Crucially, biometrics only unlock access to private keys. They never leave the device or enter cryptographic operations. Private keys and biometric templates remain protected within hardware-backed environments like the Secure Enclave (iOS) or Trusted Execution Environment/StrongBox (Android), even if the OS is compromised.

4.2. Passkeys Privacy Protections

Passkeys support GDPR principles through several privacy features. First, biometric data remains entirely on the user's device, eliminating centralized storage [32]. No global identifiers are used across services, which prevents unwanted tracking or re-identification between different RPs [32]. To further enhance privacy, Passkeys generate service-specific credentials. This makes a public–private key pair unique to the service (RP), ensuring that authentication is isolated to that domain [32]. In addition, explicit user consent is

required. This is through registration and user verification; it is required to inform users before personal data is used [32]. Passkeys operationalize GDPR principles by limiting data collection to what is strictly necessary, avoiding cross-service identifiers, and embedding user control through revocation and replacement of credentials. These privacy-by-design features reduce compliance risks for developers and service providers [32].

Passkeys, through FIDO2 standards, represent a significant advancement in secure, decentralized authentication. By combining local biometric processing, strong cryptography, hardware-backed isolation, and phishing resistance, Passkeys directly support key GDPR principles. Passkeys may avoid triggering the heightened protections under Article 9 of the GDPR. However, effective GDPR compliance depends on platform-specific implementation. Android's ecosystem faces challenges with device fragmentation and inconsistent hardware security, while iOS offers a uniform but restrictive model. Additionally, evolving spoofing tactics necessitate continuous security improvements [3,33].

5. Platform-Specific: Android Passkeys Integration

Passkeys integration on Android supports GDPR compliance by combining decentralized authentication, biometric user verification, and hardware-backed key storage. Android introduced support for the FIDO2 standard beginning with version 9.0, and later expanded its privacy and cryptographic protections in Android 11.0 and beyond, which laid the foundation for the migration to Passkeys [5,34,35]. Biometric templates and cryptographic keys are stored locally on the device, minimizing data transfer in accordance with GDPR Articles 5 and 25 [4,11]. The BiometricPrompt API, introduced in Android 9.0, standardizes biometric interactions and enforces user consent, supporting GDPR Articles 6 and 32 [4,11]. Hardware protections on Android devices include the Trusted Execution Environment (TEE) starting on 6.0+ and StrongBox on 9.0+ which isolate sensitive operations to meet GDPR technical safeguard obligations under Articles 25 and 32 [4,36,37]. There are also biometric sensors that are categorized into Class 1 (Convenience), Class 2 (Weak), and Class 3 (Strong) based on spoof resistance and pipeline security. Only Class 2 and 3 sensors receive elevated platform privileges and integration capabilities [11]. However, compliance varies by device, OS version, and manufacturer. High-end models offer stronger protections, while older or budget devices may lack adequate safeguards.

5.1. System Architecture on Android

Android's decentralized Passkeys authentication model ensures that biometric templates and cryptographic keys remain stored locally on supported devices [5]. This approach supports GDPR Articles 5 and 25 [4]. Android stores credential keys in the Keystore, isolating them from apps and external servers—an approach that supports GDPR's data minimization principle [37]. These keys remain isolated from applications, external servers, and unauthorized processes. Newer Android versions, particularly Android 11+, enhance Passkeys support with refined APIs, improved Keystore protections, and deeper platform security integration [38]. Fragmentation across the Android ecosystem introduces compliance inconsistencies. Devices running Android 9.0 or 10.0, while Passkeys-capable, may lack the latest protections for local processing and system-level security. Lower-tier models often feature outdated firmware or incomplete Passkeys functionality, undermining uniform GDPR protections. Users influence compliance through device settings and biometric data is stored locally by default, but users must enroll their biometrics through system settings. The ability to revoke biometric access by deleting templates or resetting credentials follows GDPR's user control and consent withdrawal requirements. Yet, users on outdated devices or without critical security updates may face elevated risks due to weakened local protections [39]. In summary, Android's system architecture establishes a foundation for

GDPR-compliant Passkeys deployment through decentralized key management and strict local storage. However, consistent protection depends on device capabilities, software version, and user engagement with the biometric settings on the specific device.

5.2. Application Programming Interfaces (APIs) on Android

The BiometricPrompt API, introduced in Android 9.0 (API level 28), standardizes biometric authentication interactions and reduces fragmentation [40]. BiometricPrompt abstracts biometric operations, enabling apps to request authentication while preventing direct access to biometric templates, supporting GDPR data minimization [4]. Supported modalities depend on hardware and may include fingerprint, face, or iris recognition [11]. In the Passkeys framework, BiometricPrompt works alongside the Android Passkeys API to ensure user presence verification before cryptographic keys stored in the Keystore are accessed to sign authentication challenges issued by Relying Parties (RPs) [37,41]. BiometricPrompt standardizes authentication across apps while abstracting sensitive operations away from developers, reducing exposure of biometric templates. BiometricPrompt UI and AuthService handle user-facing prompts and initial verification logic. BiometricService and FingerprintService interface with secure hardware, including biometric sensors, TEE/SE libraries, and the Keystore. Devices running Android 8 or earlier relied on the deprecated FingerprintManager, which lacked modern privacy safeguards [11]. BiometricPrompt ensures raw biometric templates are never exposed to applications; only success or failure status is returned, preserving user privacy.

5.3. Hardware Security on Android

Android Passkeys protections depend on hardware-backed components. Most modern devices use ARM TrustZone-based TEEs to isolate cryptographic operations and biometric data, inaccessible even if the OS is compromised [15]. Private keys for Passkeys authentication are generated and stored within the Keystore, with access restricted to the TEE. Templates collected during enrollment or authentication are processed within the TEE, preventing unauthorized access. Biometric sensors are categorized on Class 1, 2, 3 giving different levels of security for biometric processing [11]. Introduced with Android 9.0, the StrongBox enhances TEE security on supported devices with Hardware Security Modules (HSMs) [42]. StrongBox offers tamper-resistant key storage working with GDPR technical safeguard requirements under Articles 25 and 32 [4]. However, StrongBox is primarily present in premium devices. Many lower-cost models lack dedicated HSMs, reducing physical tamper resistance.

High-end Android devices demonstrate strong GDPR compliance with Passkeys implementation due to newer hardware and updated API support. However, platform fragmentation, hardware variation, and inconsistent developer practices continue to undermine consistent protection across the Android OS. For GDPR-sensitive Passkeys deployment, relying parties must validate device capabilities and require confirmation of device compliance.

6. Platform-Specific: iOS Passkeys Integration

Apple's iOS also delivers support for FIDO2-based Passkeys authentication while also introducing its proprietary passkey feature through iCloud Keychain. To avoid confusion, it is important to note the difference between FIDO2/WebAuthn Passkeys, standards-based credentials designed to remain local to a single device, and Apple passkeys, synchronization on credentials across devices, with iCloud Keychain. This distinction matters for GDPR compliance, since FIDO2 Passkeys are designed to remain local on the device, while Apple's iCloud-based passkeys introduce an additional layer of cross-device storage and synchronization.

Support for Passkeys on iOS has expanded progressively. External FIDO2 security keys in versions 13.3 were added WebAuthn with TouchID and FaceID in Safari in iOS 14.0, working with functionality through Apple's iCloud keychain with iOS 16.0 [22,23,43]. On iOS 16 and later, Passkeys are supported through Safari's WebAuthn, enabling passwordless authentication with Face ID, Touch ID or external security keys that are compatible with iPhones [12,44]. This functionality is supported on all modern iPhones with the Secure Enclave, including the iPhone 5s and newer for basic WebAuthn [19]. Biometric templates and cryptographic keys are securely stored on-device within the Secure Enclave, minimizing data transfer and satisfying GDPR Articles 5 and 25 [4]. Authentication is managed through the LocalAuthentication framework, which mediates biometric verification and requires explicit user interaction before credentials can be accessed. These design choices follow GDPR Articles 6 and 32 [4]. Full Passkey synchronization is available on iPhones through iCloud Keychain [45]. While this introduces convenience and cross-device login capabilities, it can also raise potential concerns under the GDPR. Particularly around data minimization and user control, since credentials are no longer strictly bound to a single device [46]. Despite this, iOS maintains consistent protections across supported models due to its tightly integrated hardware–software architecture, reducing fragmentation risks and enforcing uniform application of security safeguards.

6.1. System Architecture on iOS

iOS supports decentralized authentication by keeping biometric templates and cryptographic keys stored locally in the Secure Enclave, never transmitting them to external servers. During Passkeys credential registration, private keys are generated and securely stored within the Secure Enclave, isolated from apps and the main OS. The Secure Enclave processes, encrypts, and stores the corresponding Face ID, and Touch ID template data [12]. iOS's uniform hardware–software integration ensures consistent protection across all supported devices. In addition to native support, Apple Safari allows the use of external Passkeys security keys via NFC, Lightning, or USB [20]. Biometric enrollment is user-driven, with system settings allowing users to revoke access, delete templates, or disable biometrics, following with GDPR Article 6 [4,47]. However, iOS 16 introduced Passkey synchronization via iCloud Keychain. While this enhances convenience, it raises concerns under GDPR, particularly with regard to data minimization and user autonomy. Once stored in iCloud, synced passkeys on the Keychain are no longer strictly bound to a single device [45]. Additionally, iOS and iPadOS authenticators do not support linking for cross-device authentication. When not cross-linked, users must scan a QR code each time they authenticate across devices [43]. Improper configuration, such as weak iCloud security or failure to delete synced credentials, can undermine local-only protections and weaken GDPR safeguards.

6.2. Application Programming Interfaces (APIs) on iOS

The LocalAuthentication framework, available on iOS 8.0 and later, standardizes biometric and passcode-based authentication across Apple devices [18]. This API abstracts biometric operations, enabling applications to request Face ID, or Touch ID verification without accessing biometric templates. The API tells the user why the app requires authentification, the framework then coordinates with the Secure Enclave to carry out the operation, afterward, you receive only a Boolean result indicating authentication success or failure [18]. Supported modalities include Face ID and Touch ID, depending on device hardware. In the Passkeys framework, iOS primarily enables authentication through the WebAuthn API in Safari, which facilitates challenge-response interactions using public key cryptography. When users access a Passkeys enabled website in Safari, the browser uses

LocalAuthentication, requiring biometric verification before the Secure Enclave signs the challenge with the private key stored locally. The authenticator, generates a public–private key pair at account creation time, and sends the public key to the server. The server, known as the relying party, holds the public key for authentication, and uses assertion to challenge the authenticator to prove its identity is valid [48]. The biometric authentication and credential signing process is tightly secured across system layers. An application initiates an authentication request through the LocalAuthentication framework. LocalAuthentication, part of the iOS operating system, manages secure prompts and ensures user presence through Face ID or Touch ID. The Secure Enclave processes the biometric input and signs authentication challenges internally using stored credentials. Only a success or failure result is returned to the app, biometric templates and private keys never leave the Secure Enclave.

Passkeys and WebAuthn functionality are natively supported only in Safari and ASWebAuthentication-Session (System WebView) and full app-level integration using ASAuthorizationPlatformPublicKeyCredential-Provider is only available on iOS 15.0 and later [49,50]. Apps targeting versions earlier than iOS 15.0 cannot implement native passkey/password support, limiting their ability to offer GDPR compliant privacy controls under Article 5 and 32 [4]. Also WebViews would lack full WebAuthn support, restricting usage to the app's own domain [51]. This platform constraint prevents federated sign-in and reduces flexibility for third-party developers. From iOS 13.0 onward, WebAuthn became available in Safari, and iOS 16.0 introduced synced passkeys with iCloud Keychain sync and Secure Enclave protection, enabling cross-device credential access [45].

6.3. Hardware Security on iOS

iOS secures Passkeys credentials and biometric data through dedicated hardware measures. The Secure Enclave isolates key storage, biometric processing, and sensitive operations from the main OS. The Secure Enclave employs encrypted memory, secure boot, and hardware random number generation to protect data, even if iOS is compromised [19]. Biometric Sensor Protections: Face ID, and Touch ID utilize advanced liveness detection and store templates exclusively within the Secure Enclave, safeguarding against unauthorized access [19]. Passkeys credentials can be synchronized across Apple devices using iCloud Keychain. This synchronization involves keychain syncing and keychain recovery that holds strong privacy and security measures for data handling [52]. While this enhances usability across devices, it introduces considerations around data stored locally on device and security safeguards that may not comply with GDPR Articles 5, 6, and 32 [4]. External security key support, in addition to on-device biometrics, iOS supports external FIDO2-certified security keys through NFC, USB-C, and Lightning connectors. These external authenticators can be used as part of the WebAuthn framework within Safari or supported apps. They operate independently of the Secure Enclave, performing cryptographic operations on-device and never exposing private keys to the iOS system, maintaining strong data minimization and isolation in line with GDPR Articles 5 and 25 [4,53].

iOS offers strong GDPR-compliant Passkeys protections through local biometric processing, hardware isolation through the Secure Enclave, and a uniform system architecture. However, platform restrictions, like limited third-party integration and mandatory use of Safari or specific APIs may constrain developer flexibility and user control. Additionally, iCloud-based passkey synchronization introduces GDPR risks related to data minimization and cross-device exposure.

7. Comparative Evaluation of Passkeys on Android and iOS for GDPR Compliance

While both Android and iOS are GDPR compliant through decentralized, on-device biometric processing, the key issue is not whether they follow the regulation in theory, but how securely they implement it in practice. Both platforms minimize off-device data transfer by keeping biometric data local, and each uses hardware-backed protections to isolate biometric templates and cryptographic keys. However, the level of protection varies by platform and device, which is where the core differences emerge under closer evaluation.

7.1. Legal Evaluation

7.1.1. Article 5: Local-Only Processing

Android supports local-only biometric and cryptographic key storage through the Keystore and Trusted Execution Environment (TEE), ensuring that sensitive data remains on-device and is isolated from the main operating system [5,11,15,37]. On premium devices, the inclusion of StrongBox enhances data protection by providing tamper resistance and secure key storage in compliance with GDPR's data minimization principles [36,42]. However, due to platform fragmentation, many lower-tier or outdated devices lack StrongBox or advanced TEE implementations, leading to inconsistent enforcement of local-only processing safeguards across Android devices [11,38,39]. iOS delivers consistent local-only storage and biometric isolation through the Secure Enclave across all supported devices [19]. Uniform hardware–software integration ensures strict data minimization with reduced exposure to external systems [12]. However, iCloud Keychain synced passkeys can introduce some risk, as credentials may be replicated across multiple devices, potentially weakening the single-device principle [45,46].

7.1.2. Article 6: User Consent and Control

The BiometricPrompt API, introduced in Android 9.0 (API level 28), standardizes user authentication flows and enforces explicit user consent before any biometric operation is performed [10,11,40]. Users are provided with mechanisms to revoke access by deleting enrolled biometric data or disabling biometric authentication through system settings [11]. However, fragmented implementation across manufacturers and the continued use of older methods like FingerprintManager on older devices weakens enforcement and consistency of consent handling [11,38,39]. As a result, while user control is supported, it is not uniformly guaranteed across all Android devices. The LocalAuthentication framework with iOS enforces user interaction prior to any biometric access [18]. Users must explicitly approve biometric use during each authentication attempt and retain full control over disabling or deleting biometric data in system settings [47]. iOS devices maintain consistent enforcement of consent mechanisms across supported models due to uniform hardware–software integration [12]. The use of iCloud Keychain for passkey synchronization may introduce consent transparency concerns, as credentials can reside on multiple devices without explicit user awareness [45,46].

7.1.3. Article 25: System Design and Privacy-by-Default

Android integrates privacy-by-design through the Trusted Execution Environment (TEE), which isolates cryptographic operations and biometric data from the main OS, and through BiometricPrompt, which abstracts biometric processes away from applications [11,15]. On supported high-end devices, StrongBox adds hardware-enforced protections enhancing privacy-by-default measures [42]. Despite these features, platform fragmentation and inconsistent implementation across manufacturers reduce the effectiveness of privacy safeguards on mid- to low-tier devices [11,39]. Thus, while privacy-

by-default is well-supported on flagship models, it is not uniformly enforced across all Android devices. Apple's vertically integrated architecture applies privacy-by-design principles consistently. The Secure Enclave isolates all sensitive operations through encrypted memory and hardware-based key generation [19]. APIs such as LocalAuthentication take biometric processes and prevent app-level access to biometric templates [18]. Passkeys are origin-bound and cryptographic signing is restricted to the relying party's origin, ensuring strict scoping and minimizing false use [47]. System-level prompts further enforce user control and prevent unauthorized access to credentials [12].

7.1.4. Article 32: Technical Safeguards

High-end Android devices utilize the Trusted Execution Environment (TEE) and StrongBox hardware security modules to ensure secure key management, tamper resistance, and advanced liveness detection [11,15,42]. These features strengthen data confidentiality and security. However, lower-cost models often lack StrongBox or implement weaker TEE configurations, leading to inconsistent technical safeguards across the Android ecosystem [11,39]. iOS uses the Secure Enclave, a dedicated coprocessor with encrypted memory and hardware-based random key generation to isolate biometric processing and protect sensitive operations [19]. Face ID and Touch ID, employ advanced liveness detection, with biometric templates stored exclusively within the Secure Enclave to prevent unauthorized access [19]. Additionally, iCloud Keychain enables synchronization of passkeys across devices, resulting in credentials are stored on multiple devices [46,52].

7.2. Mobile Devices: Strengths and Limitations

A visual comparison of Android and iOS passkey implementations is provided in Table 1, serving as the basis for the following discussion of strengths and limitations. Android's strengths lie in its support for local-only biometric and key storage, which complies with GDPR's data minimization principles. High-end devices offer strong protections with the TEE and StrongBox, providing tamper-resistant key management and stronger isolation of biometric authentication. Android's BiometricPrompt API enforces explicit consent and user control and standardized authentication across applications. Android limitations include ecosystem fragmentation which leads to inconsistent GDPR compliance. Some lower-cost devices may lack StrongBox or advanced biometric protections. Also, developer-side enforcement of best practices may vary.

Apple's iOS by contrast, offers more consistency across supported devices through its Secure Enclave implementation. This provides tamper-resistant storage and strong isolation of biometric data. Local-only processing of biometric templates supports privacy-by-design, while origin-bound credentials and liveness detection enhance protection against misuse and spoofing. iOS still has its limitations which include limited third-party access restricts Passkeys adoption beyond Apple-approved apps. Additionally, iOS closed-source design limits transparency into internal system operations for these third-parties. Finally, the iCloud Keychain synchronization across connected devices may be convenient, but it introduces risks for data leakage.

While platform design establishes the baseline for biometric security, GDPR compliance ultimately lies with the data controller or the Relying Party (RP). Application developers and service providers must ensure that user authentication on their platforms meets legal obligations, regardless of underlying OS differences. On Android, fragmentation introduces challenges. Many devices lack StrongBox or employ weaker biometric sensors (Class 1 or Class 2). Developers must decide whether to allow authentication on lower-security devices, potentially increasing compliance risks. Blocking devices enhances compliance with GDPR's Article 25 (privacy-by-design), but at the cost of excluding

significant user populations in emerging markets, creating tension between accessibility and compliance. Or block unsupported models, which may reduce usability but better align with GDPR's principles of privacy-by-design (Article 25) and security of processing (Article 32). On iOS, Apple's Secure Enclave provides hardware-based isolated key storage and has been evaluated at CC EAL 5+. This design reduces variability across devices. However, relying parties remain responsible for managing consent, ensuring transparency, and explaining whether iCloud passkeys are stored locally or synchronized through iCloud. In both ecosystems, the GDPR requires relying parties to maintain accountability (Article 5(2)) and to implement organizational measures that verify device capabilities and mitigate risks. This review therefore contributes by clarifying the consensus that local-only biometric processing enhances GDPR alignment.

Table 1. Comparative Evaluation of Android and iOS Passkeys Implementations under GDPR.

GDPR Principle	Android (TEE/StrongBox, BiometricPrompt)	iOS (Secure Enclave, LocalAuthentication)
Article 5—Data Minimization	Keystore/TEE ensure local-only key storage; StrongBox (premium models) adds tamper resistance [5,11,15,36,37,42]. Fragmentation means weaker or absent protections on low-tier devices [38,39].	Secure Enclave provides consistent local-only storage and isolation [12,19]. iCloud Keychain sync replicates credentials across devices, reducing strict minimization [45,46].
Article 6—Consent & Control	BiometricPrompt enforces explicit consent and revocation [10,11,40]. Older devices using FingerprintManager provide weaker safeguards [38,39].	LocalAuthentication requires explicit approval and consistent consent handling [12,18,47]. iCloud synchronization may reduce transparency when credentials persist across devices [45,46].
Article 25—Privacy by Design/Default	TEE isolates biometric and cryptographic operations; StrongBox strengthens protections on supported devices [11,15,42]. Fragmentation undermines consistent enforcement [11,39].	Secure Enclave enforces privacy-by-design across all models [19]; APIs prevent app-level template access [18]. Passkeys are origin-bound, limiting scope and misuse [12,47].
Article 32—Security of Processing	TEE + StrongBox provide secure key management, liveness detection, and tamper resistance [11,15,42]. Lower-cost devices lack StrongBox or rely on weaker TEEs [39].	Secure Enclave provides encrypted memory, hardware random numbers, and strong liveness detection [19]. Face/Touch ID templates stay isolated; iCloud sync stores credentials across devices [46,52].

This study is descriptive and relies on a legal-technical review of platform documentation. This analysis is narrative and descriptive, without empirical testing, which constrains the extent to which compliance can be validated in practice. Future work should incorporate empirical validation, including developer case studies and device-level testing of Passkeys implementations. Such approaches would allow direct observation of consent flows, device capabilities, and RP-level strategies, complementing the descriptive findings presented here. There is a strong consensus in the literature that on-device biometric isolation reduces GDPR exposure, while areas such as Android's fragmented device ecosystem and Apples's iCloud-synced passkeys remain contested and require further empirical study. Such empirical work would confirm platform-level GDPR compliance in practice and provide evidence-based guidance for policymakers and service providers. At the same time, this synthesis clarifies where consensus across prior research is strong and where further empirical inquiry is still required.

8. Conclusions

This comparative paper demonstrates that both Android and iOS implement Passkeys with GDPR principles, Article 5 (data minimization), Article 6 (lawful processing and consent), Article 25 (privacy-by-design), and Article 32 (security of processing). Both platforms adopt a

local-only biometric processing model and use hardware-backed key isolation to minimize off-device data exposure and strengthen privacy protections [4,11,12,15,16].

Android provides strong support for GDPR- Passkeys functionality on premium devices through the Trusted Execution Environment (TEE) and StrongBox [15,16,36,42]. Additionally, the BiometricPrompt API enforces user consent, abstracting biometric interactions and enabling revocation of access through system settings [11,40]. However, Android's open ecosystem remains fragmented, many mid- and lower-tier models lack StrongBox or use varied TEE implementations, creating inconsistencies in privacy enforcement and technical safeguards [11,38,39]. Developers must address these through device verification and validation.

iOS, by contrast, offers a more consistent and tightly integrated Passkeys implementation. The Secure Enclave, standard across supported devices, ensures encrypted memory, hardware-rooted key generation, and isolated biometric processing [19]. iOS's LocalAuthentication framework provides consistent consent enforcement, origin-bound credentials, and protection against unauthorized biometric access at the app level [12,18,47,48]. However, iOS's closed-source nature limits external apps, and the introduction of iCloud Keychain passkey synchronization introduces potential risks to data minimization and user control. Replicating credentials across devices may reduce transparency.

Both platforms implement spoof resistance and liveness detection, helping to protect against biometric attacks [3,33]. Nevertheless, biometric authentication systems still carry inherent risks, such as irrevocability if compromised, the potential for unauthorized surveillance, and the challenges of classification under GDPR Article 9 if linked to broader identity systems [1,3,27–29]. Local-only biometric processing, when isolated and unlinked to external identifiers, may avoid classification as special category data. Reinforcing the importance of system architecture and context [28,29].

Compliance with GDPR in the context of Passkeys is not solely a platform issue. The Relying Party bears direct responsibility for assessing device-level protections, managing consent, and ensuring that users on weaker devices are not exposed to disproportionate risk. Future work should further explore RP-level strategies to enhance GDPR compliance. Such as device compliance checks, adaptive authentication policies, and transparent user communication, to better support GDPR compliance across diverse mobile ecosystems.

Ultimately, Passkeys represent a significant step forward in privacy-preserving, passwordless authentication. Its decentralized, cryptographic model supports GDPR Articles for biometric security, but real-world compliance varies based on platforms and devices. Service providers and developers must continue to evaluate platform-level protections, enforce best practices, and ensure standards to fully realize Passkeys' potential under the GDPR.

Based on our analysis, several guidelines can help strengthen GDPR compliance in Passkey deployment across platforms. Developers and relying parties should verify device capabilities before enabling Passkeys. They should not use Android Class 1 or Class 2 biometrics, since these fall short of the FIDO Alliance's definition of a restricted operating environment. Service providers should maintain up-to-date compliance information on supported devices and implement fallback authentication methods when protections vary between platforms or device models. Policymakers can promote greater consistency by clarifying GDPR expectations around cloud-synced credentials and encouraging certification baselines that address both Android fragmentation and the closed ecosystem of iOS.

Existing research gaps highlight the need for empirical validation, including device-level testing of Android models with varying hardware, evaluation of iCloud-based credential syncing under GDPR, and user studies on consent and control. Future studies should therefore move beyond documentation analysis to include experimental testing and field evaluations. This can validate whether Android's fragmented ecosys-

tem and Apple's iCloud synced passkeys implementation meet GDPR obligations in real-world deployments.

Author Contributions: Background research with primary study including drafting the original manuscript was performed by A.C. The overall conceptual foundation for the comparative analysis and reflective guidance on the overall argument was provided by S.L.; who also supervised the work. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Data Availability Statement: No specific dataset was used. The pertinent information is properly cited throughout the manuscript.

Acknowledgments: Joshua Schoetker provided revisions and refinements that improved the clarity and structure of the paper.

Conflicts of Interest: The authors declare no conflicts of interest.

Abbreviations

API	Application Programming Interface
CTAP2	Client to Authenticator Protocol v2
EEA	European Economic Area
EU	European Union
GDPR	General Data Protection Regulation
HSM	Hardware Security Module
iOS	iPhone Operating System
NFC	Near Field Communication
PIN	Personal Identification Number
RP	Relying Party
TEE	Trusted Execution Environment
WebAuthn	Web Authentication
W3C	World Wide Web Consortium

References

1. Das, A.; Galdi, C.; Han, H.; Ramachandra, R.; Dugelay, J.-L.; Dantcheva, A. Recent Advances in Biometric Technology for Mobile Devices. 2018. Available online: https://ieeexplore.ieee.org/document/8698587 (accessed on 26 June 2025).
2. Stragapede, G.; Vera-Rodriguez, R.; Tolosana, R.; Morales, A.; Acien, A.; Le Lan, G. Mobile Behavioral Biometrics for Passive Authentication. *Pattern Recognit. Lett.* **2022**, *157*, 35–41. [CrossRef]
3. Malik, G. Biometric Authentication-Risks and Advancements in Biometric Security Systems. *J. Comput. Sci. Technol. Stud.* **2024**, *6*, 159–180. [CrossRef]
4. GDPR. General Data Protection Regulation—Official Legal Text. 2016. Available online: https://gdpr-info.eu/ (accessed on 9 June 2025).
5. FIDO Alliance. Passkeys. 2025. Available online: https://fidoalliance.org/passkeys/ (accessed on 2 August 2025).
6. FIDO Alliance. Passkeys: Specifications Overview. 2025. Available online: https://fidoalliance.org/specifications-overview/ (accessed on 2 August 2025).
7. FIDO Alliance. Client to Authenticator Protocol (CTAP). 2019. Available online: https://fidoalliance.org/specs/fido-v2.0-ps-20190130/fido-client-to-authenticator-protocol-v2.0-ps-20190130.html (accessed on 9 June 2025).
8. FIDO Alliance. Sign in With Passkey. 2023. Available online: https://www.passkeycentral.org/design-guidelines/required-patterns/sign-in-with-a-passkey (accessed on 15 June 2025).
9. Lyastani, S.G.; Schilling, M.; Neumayr, M.; Backes, M.; Bugiel, S. Is FIDO2 the Kingslayer of User Authentication? A Comparative Usability Study of FIDO2 Passwordless Authentication. 2020. Available online: https://ieeexplore.ieee.org/document/9152694 (accessed on 9 June 2025).
10. FIDO Alliance. FAQ on FIDO Relevance for the GDPR. 2018. Available online: https://fidoalliance.org/wp-content/uploads/FIDO_Alliance_GDPR_FAQ_September2018.pdf (accessed on 15 June 2025).

11. Android Open Source Project. Biometric Security Features. 2025. Available online: https://source.android.com/docs/security/features/biometric (accessed on 16 June 2025).
12. Apple Inc. Biometric Security. 2025. Available online: https://support.apple.com/en-ca/guide/security/sec067eb0c9e/web (accessed on 11 June 2025).
13. FIDO Alliance. FIDO Alliance White Paper. 2022. Available online: https://fidoalliance.org/wp-content/uploads/2022/12/FIDO-e-Government-White-Paper.pdf (accessed on 15 June 2025).
14. Kepkowski, M.; Machulak, M.; Kaafar, D. Challenges with Passwordless FIDO2 in an Enterprise Setting: A Usability Study. In Proceedings of the IEEE Secure Development Conference, Atlanta, GA, USA, 18–20 October 2023; pp. 37–48. Available online: https://www.computer.org/csdl/proceedings-article/secdev/2023/313200a037/1RVZwakmGeA (accessed on 20 June 2025).
15. Android Open Source Project. Trusty TEE. 2025. Available online: https://source.android.com/docs/security/features/trusty (accessed on 16 June 2025).
16. Android Open Source Project. Hardware Security Best Practices. 2025. Available online: https://source.android.com/docs/security/best-practices/hardware (accessed on 16 June 2025).
17. Authgear. How Biometric Authentication Works? Unlock the Future of Security. 2025. Available online: https://www.authgear.com/post/how-does-biometric-authentication-work-a-comprehensive-guide-to-the-future-of-security (accessed on 16 June 2025).
18. Apple Inc. Local Authentication. 2025. Available online: https://developer.apple.com/documentation/localauthentication (accessed on 11 June 2025).
19. Apple Inc. Secure Enclave. 2025. Available online: https://support.apple.com/en-ca/guide/security/sec59b0b31ff/web (accessed on 11 June 2025).
20. TNW. Safari Gets Support for Hardware Security Keys with iOS 13.3. 2019. Available online: https://thenextweb.com/news/safari-gets-support-for-hardware-security-keys-with-ios-13-3 (accessed on 13 June 2025).
21. FIDO Alliance. Expanded Support for FIDO Authentication in iOS and MacOS. 2020. Available online: https://fidoalliance.org/expanded-support-for-fido-authentication-in-ios-and-macos/ (accessed on 20 June 2025).
22. Yubico. Security Key Compatibility. 2025. Available online: https://developers.yubico.com/WebAuthn/Supporting_U2F_or_FIDO2_Security_Keys_on_iOS_or_iPadOS/Security_Key_Compatibility.html (accessed on 20 June 2025).
23. Apple Inc. About Security Keys for Apple Account. 2024. Available online: https://support.apple.com/en-us/102637 (accessed on 11 June 2025).
24. Token2. Native Support of NFC FIDO2 Keys in iPhone Safari. 2025. Available online: https://www.token2.com/shop/page/native-support-of-nfc-fido2-keys-in-iphone-safari (accessed on 11 June 2025).
25. StatCounter. Mobile Operating System Market Share Worldwide. 2025. Available online: https://gs.statcounter.com/os-market-share/mobile/worldwide (accessed on 20 June 2025).
26. FIDO Alliance. FIDO UAF Architectural Overview. 2014. Available online: https://fidoalliance.org/specs/fido-uaf-v1.0-ps-20141208/fido-uaf-overview-v1.0-ps-20141208.html (accessed on 15 June 2025).
27. Zhang, J.; Liu, Z.; Luo, X. Unraveling juxtaposed effects of biometric characteristics on user security behaviors: A controversial information technology perspective. *Decis. Support Syst.* **2024**, *183*, 114267. [CrossRef]
28. Yang, Q.; Lepore, C.; Eynard, J.; Laborde, R. From theory to practice: Data minimization and technical review of verifiable credentials under the GDPR. *Comput. Law Secur. Rev.* **2025**, *57*, 106138. [CrossRef]
29. Bisztray, T.; Gruschka, N.; Bourlai, T.; Frotsch, L. Emerging Biometric Modalities and their Use: Loopholes in the Terminology of the GDPR and Resulting Privacy Risks. In Proceedings of the International Conference of the Biometrics Special Interest Group (BIOSIG), Darmstadt, Germany, 15–17 September 2021; pp. 1–5. Available online: https://ieeexplore.ieee.org/abstract/document/9548298 (accessed on 16 June 2025).
30. World Wide Web Consortium. Web Authentication: An API for Accessing Public Key Credentials. 2021. Available online: https://www.w3.org/TR/webauthn-2/ (accessed on 15 June 2025).
31. FIDO Alliance. Download Authentication Specifications. 2025. Available online: https://fidoalliance.org/specifications/download/ (accessed on 2 August 2025).
32. FIDO Alliance. FIDO Alliance Privacy Principles. 2025. Available online: https://fidoalliance.org/fido-authentication-2/privacy-principles/ (accessed on 15 June 2025).
33. Balamurugan, M. Biometric Authentication: A Double-Edged Sword for Security. *Int. J. Sci. Res.* **2024**, *13*, 170–173.
34. FIDO Alliance. Android Now FIDO2 Certified, Accelerating Global Migration Beyond Passwords. 2019. Available online: https://fidoalliance.org/android-now-fido2-certified-accelerating-global-migration-beyond-passwords/ (accessed on 10 June 2025).
35. TechTarget. Android 11 Features Zero in on Security, Privacy. 2020. Available online: https://www.techtarget.com/searchmobilecomputing/news/252479809/Android-11-features-zero-in-on-security-privacy (accessed on 20 June 2025).
36. Android Open Source Project. Android Compatibility Definition Document. 2025. Available online: https://source.android.com/docs/compatibility/cdd (accessed on 20 June 2025).

37. Android Open Source Project. Android Keystore System. 2025. Available online: https://developer.android.com/privacy-and-security/keystore (accessed on 20 June 2025).
38. Android Open Source Project. Privacy in Android 11. 2024. Available online: https://developer.android.com/about/versions/11/privacy (accessed on 20 June 2025).
39. Sasi, T.; Lashkari, A.; Lu, R.; Xiong, P.; Iqbal, S. A comprehensive survey on IoT attacks: Taxonomy, detection mechanisms and challenges. *J. Inf. Intell.* **2024**, *2*, 455–513.
40. Android Open Source Project. System Security Best Practices. 2025. Available online: https://source.android.com/docs/security/best-practices/system (accessed on 20 June 2025).
41. Google Identity. FIDO2 API for Android. 2025. Available online: https://developers.google.com/identity/fido/android/native-apps (accessed on 11 June 2025).
42. Android Open Source Project. StrongBox KeyMint. 2025. Available online: https://developer.android.com/privacy-and-security/keystore#StrongBoxKeyMint (accessed on 20 June 2025).
43. Passkeys. Resources for Passkeys in Apple's iOS and iPadOS. 2025. Available online: https://passkeys.dev/docs/reference/ios/ (accessed on 20 June 2025).
44. Passkeys. What Is a FIDO Passkey? FIDO Passkey Explained. 2025. Available online: https://www.passkeys.com/fido-passkey (accessed on 2 August 2025).
45. Apple Inc. Set Up iCloud Keychain. 2024. Available online: https://support.apple.com/en-us/109016 (accessed on 20 June 2025).
46. Apple Inc. About the Security of Passkeys. 2024. Available online: https://support.apple.com/en-us/102195 (accessed on 11 June 2025).
47. Apple Inc. Face ID & Privacy. 2025. Available online: https://www.apple.com/legal/privacy/data/en/face-id/ (accessed on 20 June 2025).
48. Apple Inc. Public-Private Key Authentication. 2025. Available online: https://developer.apple.com/documentation/authenticationservices/public-private-key-authentication (accessed on 20 June 2025).
49. Apple Inc. ASWebAuthenticationSession. 2025. Available online: https://developer.apple.com/documentation/authenticationservices/aswebauthenticationsession (accessed on 20 June 2025).
50. Apple Inc. ASAuthorizationPlatformPublicKeyCredentialProvider. 2025. Available online: https://developer.apple.com/documentation/authenticationservices/asauthorizationplatformpublickeycredentialprovider (accessed on 20 June 2025).
51. Passkeys Glossary. iOS-Implementation for Flutter iOS. 2024. Available online: https://passkeys-auth.com/docs/implementation/flutter/ios/ (accessed on 20 June 2025).
52. Apple Inc. iCloud Keychain Security Overview. 2024. Available online: https://support.apple.com/guide/security/icloud-keychain-security-overview-sec1c89c6f3b/web (accessed on 21 June 2025).
53. Yubico. WebAuthn Walk-Through. 2025. Available online: https://developers.yubico.com/WebAuthn/WebAuthn_Walk-Through.html (accessed on 20 June 2025).

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.